aa1

PASSING INTO OPERATION SYSTEM MESSAGE PASSING USING PROXY OBJECTS--.

IN THE DRAWINGS

Applicant submits herewith Figures 1, 6, 7, and 8B with corrections indicated in red. Applicant will submit formal drawings upon receipt of a notice of allowance.

IN THE CLAIMS

Sub E1

1. (Once amended) A method for sending [a] an object oriented programming language based message having dynamic binding from a first object in a first process to a second object in a second process, said method comprising the steps of:

a1

providing, using a first processing means, said object oriented programming language based message to a first proxy in said first process;

using said first proxy and said first processing means, encoding said object oriented programming language based message into an operating system based message at run time;

transmitting said operating system based message to said second process at run time;

decoding, using a second processing means, said operating system based message into a language based message;

providing, using said second processing means, said object oriented programming language based message to said second object.

2. (Once amended) The method of claim 1 further including the steps of:

said second object executing said object oriented programming language based message, using said second processing means, and generating [a] an object oriented programming language based result;

encoding, using said second processing means, said object oriented programming language based result into an operating system based result at run time;

a1

transmitting, using said second processing means, said operating system based result to said first process at run time;

decoding said operating system based result into [a] an object oriented programming language based result at run time, using said first processing means;

providing, using said first processing means, said object oriented programming language based result to said first object.

3. (Once amended) The method of claim 1 wherein said object oriented programming language based message comprises a method and an argument.

5. (Once amended) The method of claim 2 further including the steps of:

a2

said second object generating, using said second processing means, [a] an object oriented programming language based query;

encoding, using said second processing means, said object oriented programming language based query into an operating system based query at run time;

transmitting said operating system based query to said first process at run time, using said second processing means;

decoding, using said first processing means, said operating system based query into [a] an object oriented programming language based query at run time;

a 2

providing, using said first processing means, said object oriented programming language based query to said first object.

6. (Once amended) The method of claim 5 further including the steps of:

said first object generating, using said first processing means, [a] an object oriented programming language based reply;

encoding said object oriented programming language based reply into an operating system based reply at run time, using said first processing means;

transmitting, using said first processing means, said operating system based reply to said second process at run time;

decoding, using said second processing means, said operating system based reply into [a] an object oriented programming language based reply at run time;

providing, using said second processing means, said object oriented programming language based reply to said second object.

8. (Once amended) The method of claim 1 wherein said object oriented programming language based message comprises an objective C message.

a3

Sub B3

11. (Once amended) A method for sending [a] an object oriented programming language based message having dynamic binding from a first object in a first process to a second object in a second process, said method comprising the steps of:

a4

providing, using a first processing means, said object oriented programming language based message to a first proxy in said first process;

using said first proxy and said first processing means, encoding said object oriented programming language based message into an operating system based message at run time;

transmitting, using said first processing means, said operating system based message to said second process at run time;

decoding, using said second processing means, said operating system based message into [a] an object oriented programming language based message at run time;

providing, using said second processing means, said object oriented programming language based message to said second object;

said second object generating [a] an object oriented programming language based query, using said second processing means;

creating, using said second processing means, a second proxy in said second process;

providing, using said second processing means, said object oriented programming language based query to said second proxy;

using said second proxy and said second processing means, encoding said object oriented programming language based query into an operating system based query at run time;

transmitting, using said second processing means, said operating system based query to said first process at run time;

a4

decoding, using said first processing means, said operating system based query into [a] an object oriented programming language based query at run time;

providing, using said first processing means, said object oriented programming language based query to said first object;

said first object generating [a] an object oriented programming language based reply, using said first processing means;

encoding, using said first processing means, said object oriented programming language based reply into an operating system based reply at run time;

transmitting, using said first processing means, said operating system based reply to said second process at run time;

decoding, using a second processing means, said operating system based reply into [a] an object oriented programming language based reply at run time;

providing, using said second processing means, said object oriented programming language based reply to said second object;

said second object executing said object oriented programming language based message, using said second processing means, and generating [a] an object oriented programming language based result;

encoding, using said second processing means, said object oriented programming language based result into an operating system based result at run time;

a 4

transmitting, using said second processing means, said operating system based result to said first process at run time;

decoding, using said first processing means, said operating system based result into [a] an object oriented programming language based result;

providing, using said first processing means, said object oriented programming language based result to said first object.

12. (Once amended) The method of claim 11 wherein said object oriented programming language based message comprises a method and an argument.

15. (Once amended) The method of claim 11 wherein said object oriented programming language based message comprises an objective C message.

a 5